جامعة المرقب / الخمس

كلية التربية - الخمس / سوق الخميس

اساسيات البرمجة بلغة الجافا

المحاضرة : الثامنة والتاسعة

المصفوفات في لغة الجافا

Arrays in Java

المحاضر : سالم مسعود الدروقي

Out lines

- ما هي المصفوفات وما هي انواعها .
- المصفوفات ذات البعد الواحد :
  - الاعلان عن مصفوفة ذات بعد واحد .
  - تخصيص القيم للمصفوفة ذات البعد الواحد .
  - قراءة وطباعة عناصر مصفوفة ذات بعد واحد .
  - امثلة عن المصفوفات ذات البعد الواحد .
- المصفوفات ذات البعدين:
  - الاعلان عن مصفوفة ذات بعدين.
  - تخصيص القيم للمصفوفة ذات بعدين.
  - قراءة وطباعة عناصر مصفوفة ذات بعدين.
  - امثلة عن المصفوفات ذات البعدين.

ما هي المصفوفات وما هي انواعها
What is Array & What is Array Types

- **المصفوفة** هي عبارة عن مجموعة من العناصر المخزنة في الذاكرة تحت اسم واحد (متغير واحد) ومن نوع واحد يمكن الوصول الى عناصرها عن طريق فهرس المصفوفة .
- تستخدم المصفوفات عند الرغبة في تخزين و معالجة كم كبير من المعلومات حيث يتم استخدام المصفوفات بدلا من استخدام عدد كبير من المتغيرات في عملية تخزين البيانات.
- **توجد هنالك عدة انواع من المصفوفات اهمها:**
- المصفوفات ذات البعد الواحد One-Dimensional Arrays
- المصفوفات ذات البعدين Two-Dimensional Arrays

المصفوفات ذات البعد الواحد(One-Dimensional Arrays)

- هي عبار عن مجموعة من العناصر (من نوع واحد) المرتبة في صف واحد او عمود واحد ولها فهرس واحد ويمكن تمثيل المصفوفة ذات البعد الواحد بالاشكال التالية :





- حيث يمثل الحرف A اسم المصفوفة و العدد 6 عدد عناصر المصفوفة.
- ويشار الى كل عنصر من عناصر المصفوفة باستخدام اسم المصفوفة ورقم العنصر (ترتيب العنصر (موقع العنصر) ابتدأ بالرقم صفر .

الاعلان عن مصفوفة ذات بعد واحد

- للاعلان عن مصفوفة ذات بعد واحد في لغة الجافا يجب اتباع الشكل التالي:

```
Type Array_Name[];
Array_Name=new type[size];
```

- ويمكننا دمج السطرين في سطر واحد كما يلي:

```
Type Array_Name[]=new type[size];
```

- حيث
- type: نوع عناصر المصفوفة.
- Array_Name: هو اسم المتغير الذي يمثل المصفوفة.
- [size]: يعبر عن عدد مواقع المصفوفة ويجب ان يكون عددا صحيحا موجبا و يوضع بين قوسين .
- new : لتحديد مواضع عناصر المصفوفة واخد نسخة جديدة من الفئة الذي تمثل المصفوفة.

تخصيص قيم لمواقع المصفوفة :

- **تخصيص هذه القيم من خلال كتابتها مباشرا في البرنامج من قبل المبرمج.**
- في هذه الحالة تكون عناصر المصفوفة ثابتة لا يمكن للمستخدم تغيرها في كل مره يتم فيها تنفيذ البرنامج.
- المثال التالي يقوم بحجز موقع لمصفوفة احادية البعد مكونة من 5 عناصر من النوع الصحيح مع

تخصيص قيم لهذه المصفوفة.

```
int arr[]=new int[5];
 arr[0]=15;
 arr[1]=0;
 arr[2]=133;
 arr[3]=6;
 arr[4]=17;
```

تخصيص قيم لمواقع المصفوفة :

- **تخصيص هذه القيم من خلال استخدام جملة الادخال:**
- توفر هذه الطريقة للمستخدم امكانية ادخال قيم جديدة في كل مره يتم فيها تنفيذ البرنامج.
- المثال التالي يقوم بحجز موقع لمصفوفة احادية البعد مكونة من 5 عناصر من النوع الصحيح مع اعطاء المستخدم امكانية ادخال قيم مواقع المصفوفة عن طريق لوحة المفاتيح:

```
Scanner input=new Scanner(System.in);
    int arr[]=new int[5];
    arr[0]=input.nextInt();
    arr[1]=input.nextInt();
    arr[2]=input.nextInt();
    arr[3]=input.nextInt();
    arr[4]=input.nextInt();
```

قراءة وطباعة عناصر المصفوفة باستخدام حلقات التكرار

- تستخدم جملة الادخال مع احدى انواع حلقات التكرار لاعطاء المستخدم امكانية ادخال عناصر المصفوفة عنصرا عنصرا وتخزينها في المواقع التي تم حجزها مسبقا في الذاكرة.
- تستخدم جملة الادخال مع احدى انواع حلقات التكرار لطباعة عناصر المصفوفة المخزنة مسبقا.
- في كلى العمليتين يمثل عداد حلقة التكرار فهرس المصفوفة.
- عملية تنفيذ جملة الادخال او الاخراج سوف تتكرر بعدد عناصر المصفوفة.

الشكل العام لقراءة وطباعة عناصر المصفوفة باستخدام حلقات التكرار

- الشكل العام لقراءة مصفوفة مكونه من عدد محدد من العناصر الصحيحة باستخدام حلقة التكرار for:

```
for(counter=0;counter<arr_size;counter++)
arr_name[counter]=input.nextInt;
```

- الشكل العام لطباعة مصفوفة مكونه من عدد محدد من العناصر الصحيحة باستخدام حلقة التكرار for
- لطباعة المصفوفة في صف واحد:

```
for(counter=0;counter<arr_size;counter++)
System.out.print(arr_name[counter])
```

- لطباعة المصفوفة في عمود واحد:

```
for(counter=0;counter<arr_size;counter++)
System.out.println(arr_name[counter])
```

اكتب برنامج لقراءة مصفوفة أحادية البعد مكونة من 50 عنصر من النوع الحقيقي وطباعة عناصرها في صف واحد.


```
import java.util.*;
public class JavaApplication9
{
public static void main(String args[])
{
Scanner input = new Scanner (System.in);
int arr[]=new int[50];
for (int i =0;i<50;i++)
    arr[i]=input.nextInt();
  for (int i =0;i<50;i++)
    System.out.println(arr[i]);
} }
```

اكتب برنامج لقراءة مصفوفة أحادية البعد مكونة من 20 عنصر وطباعة مجموع عناصرها.


```
import java.util.*;
public class JavaApplication9 {
public static void main(String args[])
{
 Scanner input = new Scanner (System.in);
 int arr[]=new int[20];
 int sum=0;
for (int i =0;i<20;i++)
   arr[i]=input.nextInt();

for (int i =0;i<20;i++)
  sum+=arr[i];
System.out.println("sum="+sum);
 }
}
```

اكتب برنامج لقراءة مصفوفة ثنائية البعد مكونة من 9 عناصر من نوع السلسلة 3×3 وطباعة عناصر القطر الرئيسي .


```
import java.util.Scanner;
 public class JavaApplication7 {
public static void main(String[] args)
{
Scanner input= new Scanner (System.in);
int  i,j ;
String arr[][]=new String[3][3];
for (i=0;i<3;i++)
for (j=0;j<3;j++)
arr[i][j]=input.next();

for (i=0;i<3;i++)
for (j=0;j<3;j++)
 if (i==j)
 System.out.print("  "+arr[i][j]);
 }
  }
```

اكتب برنامج لقراءة مصفوفة احادية البعد مكونة من 20 عنصر من النوع الصحيح وطباعة اصغر واكبر قيمة.


```
import java.util.*;
public class JavaApplication9 {
public static void main(String args[])
{
 Scanner input = new Scanner (System.in);
 int A[]=new int[12];
 int min,max;
for (int i =0;i<12;i++)
   A[i]=input.nextInt();
min=max=A[0];
for (int i =0;i<20;i++)
  if (A[i]<min)
      min=A[i];
  else if (A[i]>max)
      max=A[i];
System.out.println("min="+min);
System.out.println("max="+max);
 }}
```

اكتب برنامج يقوم بتدوين بيانات مخزن مواد غذائية به n صنف بحيث يقوم بقراءة اسم الصنف وسعره ثم طباعة سعر اغلي صنف ثم اسمه.


```
import java.util.*;
public class JavaApplication9 {
public static void main(String args[])
{
 Scanner input = new Scanner (System.in);
 String item[]=new String[150];
 float price[]=new float[150];
 float max_price;
 String max_item;
 int num;
 System.out.print(" Enter number of items");
num=input.nextInt();
for (int i =0;i<num;i++)
{
System.out.println("Enter item " + i + " name");
item[i]=input.next();
System.out.println("Enter item " + i+ " price");
price[i]=input.nextFloat();
}
max_price=price[0];
max_item=item[0];
for (int i =0;i<num;i++)
   if (price[i]>max_price)
   {
      max_price=price[i];
      max_item=item[i];
   }
System.out.println("max price="+max_price);
System.out.println("the most expencive item="+max_item);
 }
 }
```

المصفوفات ذات البعدين
Two-Dimensional Arrays

- هي عبارة عن مصفوفة على هيئة جدول مكونة من مجموعة من الصفوف والاعمدة.
- يتم قراءة عناصر هذه المصفوفة عن طريق فهرس المصفوفة المكون من متغيرين متغير للصفوف ومتغير للاعمدة.

A	0	1	2
0	A[0,0]	A[0,1]	**A[0,2]**
1	A[1,0]	A[1,1]	**A[1,2]**
2	A[2,0]	A[2,1]	**A[2,2]**

- حيث يمثل المتغير "A" اسم المصفوفة.
- ويمثل العددين [3*3] عدد الصفوف وكذلك عدد الاعمدة على التوالي(من اليسار الى اليمين)
- يشار الى كل عنصر من عناصر المصفوفة عن طريق اسم المصفوفة ثم كتابة رقم الصف ورقم العمود

الاعلان عن مصفوفة ذات بعدين

- للاعلان عن مصفوفة ذات بعدين في لغة الفجوال بيسك يجب اتباع الشكل التالي:

```
Type Array_Name[][];
Array_Name=new type[M][N];
```

ويمكننا دمج السطرين في سطر واحد كما يلي:

```
Type Array_Name[]=new type[M][N];
```

حيث

- type: نوع عناصر المصفوفة.
- Array_Name: هو اسم المتغير (المصفوفة).
- new : لتحديد مواضع عناصر المصفوفة واخد نسخة جديدة من الفئة التي تمثل المصفوفة.
- M: عدد صفوف المصفوفة .
- N: عدد اعمدة المصفوفة.

تخصيص قيم لمواقع المصفوفة :

- **تخزين القيم بكتابتها مباشرا في البرنامج:** في هذه الحالة تكون عناصر المصفوفة ثابتة لا يمكن للمستخدم تغيرها في كل مره يتم فيها تنفيذ البرنامج.

```
A[0,0]=10;
A[0,1]=50;
A[0,2]=15;
A[1,0]=3;
```

- **تخزين القيم من خلال استخدام جملة القراءة :**توفر هذه الطريقة للمستخدم امكانية ادخال قيم جديدة في كل مره يتم فيها تنفيذ البرنامج.

```
A[0,0]= input.nextInt();
A[0,1]= input.nextInt();
A[0,2]= input.nextInt();
A[1,0]= input.nextInt();
```

الشكل العام لقراءة وطباعة عناصر المصفوفة ذات بعدين

- الشكل العام لقراءة المصفوفة ذات البعدين باستخدام حلقتي **For**

```
for (Counter1 = 0 ; Counter1< nuumber_of_rows; Counter1 ++)
    for(Counter2 = 0 ; Counter2< nuumber_of_col; Counter2 ++)
     Array_Name[Counter1,Counter2]= input.nextType();
```

- لشكل العام لطباعة المصفوفة ذات البعدين باستخدام حلقتي **For**

```
for (Counter1 = 0 ; Counter1< nuumber_of_rows; Counter1 ++)
{
    for(Counter2 = 0 ; Counter2< nuumber_of_col; Counter2 ++)
    System.out.print(Array_Name[Counter1,Counter2]);
    System.out.print();
}
```

اكتب برنامج لقراءة مصفوفة ثنائية البعد مكونة من25 عنصر من النوع الصحيح 5*5 ثم طباعة عناصرها.


```
public class JavaApplication7
{
 public static void main(String[] args)
   {
 Scanner input= new Scanner (System.in);
 int x[][]=new int[5][5];
 int i,j;
 for(i=0;i<5;i++)
 for (j=0;j<5;j++)
 {
 System.out.print("Enter Item(" + i + "," + j + ")=");
           x[i][j]=input.nextInt();
           }
for(i=0;i<5;i++)
     {
      for (j=0;j<5;j++)
         {
            System.out.print(x[i][j]+"  ");
         }
         System.out.println();
     }
         }
}
```

اكتب برنامج بقراءة اسماء 5 طلاب وكذلك درجاتهم في 3 مواد ثم يقوم بطباعة اسم معدل كل طالب وكذلك طباعة اسم ومعدل الطالب المتحصل على اعلى معدل.


```
import java.util.Scanner;
public class JavaApplication7 {
public static void main(String[] args)
  {
Scanner input= new Scanner (System.in);
int  i,j ,sum; double avg,max=0;
String maxname="";
String name[]=new String[5] ;
int Grade[][]=new int[5][3];
for (i=0;i<5;i++)
{
System.out.print("Enter student " + i + " name:");
name[i]=input.next();
sum=0;
for (j=0;j<3;j++)
{
System.out.print("Enter student" + i + " grade" + j + "=");
Grade[i][j]=input.nextInt();
sum = sum + Grade[i][j]; }
avg = sum / 3;

System.out.println("Name=" + name[i]);
System.out.println ("avg=" +avg);
if(avg > max )
{
max = avg; maxname = name[i];
} }
System.out.println("MaxName:"+maxname);
  System.out.println("max="+max);
  }
}
```

اكتب برنامج لقراءة مصفوفة ثنائية البعد 3*4 وطباعة عناصر العمود الثاني.


```
import java.util.Scanner;
public class JavaApplication7 {
   public static void main(String[] args)
   {
Scanner input= new Scanner (System.in);
      int  i,j ;
      int arr[][]=new int[3][4];
      for (i=0;i<3;i++)
        for (j=0;j<4;j++)
      arr[i][j]=input.nextInt();
      for (i=0;i<3;i++)
      for (j=0;j<1;j++)
       System.out.println("  "+arr[i][1]);
      }
      }
```

اكتب برنامج لقراءة مصفوفة ثنائية البعد 5*5 وطباعة اكبر عنصر في الصف 3.


```
import java.util.Scanner;
public class JavaApplication7 {
   public static void main(String[] args)
   {
Scanner input= new Scanner (System.in);
      int  i,j,max ;
      int arr[][]=new int[5][5];
      for (i=0;i<5;i++)
        for (j=0;j<5;j++)
         arr[i][j]=input.nextInt();
        max=arr[0][0];
      for (i=0;i<1;i++)
        for (j=0;j<5;j++)
if (arr[2][j]>max)
max=arr[2][j];
System.out.print(" Max "+max);
 }
 }
```

اكتب برنامج يقوم بقراءة مصفوفة لاسماء ثلاثة طلاب ومصفوفة لدراجاتهم في ثلاثة مواد ثم يقوم بحساب معدل كل طالب وتخزينة في مصفوفة ثالثة ثم يقوم ثم يقوم بطباعة اسماء الطلاب ومعدلاتهم وكذلك اسم ومعدل الطالب المتحصل على اعلى معدل.


```
import java.util.Scanner;
public class JavaApplication7 {
  public static void main(String[] args)
  {
    Scanner input= new Scanner (System.in);
    int  i,j ,sum;       double avg,max=0;    String maxname="";
    String name[]=new String[3] ;
    int Grade[][]=new int[3][3];
    double avge[]=new double [3];
    for (i=0;i<3;i++)
    {
      System.out.print("Enter student " + i + " name:");
     name[i]=input.next();
     sum=0;
    for (j=0;j<3;j++)
      {
        System.out.print("Enter student" + i + " grade" + j + "=");
        Grade[i][j]=input.nextInt();
        sum = sum + Grade[i][j];
      }
```

تابع المثال في الشريحة السابقة


```
 avg = sum / 3;    avge[i]=avg;
}
System.out.println("name " + "|   Average");
System.out.println("--------------------");
    for (i=0;i<3;i++)
    {
    System.out.print(name[i]+ " | ");
    System.out.println(avge[i]);
    }
    for (i=0;i<3;i++)
    if (avge[i]>max)
    {max=avge[i];
    maxname=name[i];
    }
    System.out.println("--------------------");
   System.out.println ("name:" + maxname);
   System.out.println("max=" + max);
   }
}
```

النهاية

شكر لحسن استماعكم

المحاضر : سالم مسعود الدروقي

Salemaldrugi@gmail.com
Salemaldrugi@yahoo.com